УДК 004.8

Ю.В. Крак, А.С. Тернов, М.П. Лісняк
Інститут кібернетики ім. В.М. Глушкова НАН України, м. Київ, Україна
Україна, 03680, м. Київ, пр. Академіка Глушкова, 40

Розробка архітектури та основних інструментів комп'ютерної анімації для побудови системи синтезу жестової мови

Iu.V. Krak, A.S. Ternov, M.P. Lisniak
V.M. Glushkov Cybernetics Institute of NASU, Ukraine
Ukraine, 03680, c. Kyiv, Akademika Glushkova av.,40

The Development of the Architecture and Basic Implements of the Computer Animation for Sign Language System Synthesis Creation

Ю.В. Крак, А.С. Тернов, М.П. Лисняк
Институт кибернетики имени В.М. Глушкова НАН Украины, Украина
Украина, 03680, г. Киев, пр. Академика Глушкова, 40

Разработка архитектуры и основных инструментов компьютерной анимации для построения системы синтеза жестового языка

У статті пропонується підхід до розробки програмної системи для синтезу та анімації жестів, емоційно-мімічних проявів на обличчі людини з використанням тривимірних моделей тіла людини. Описуються основні вимоги до системи, проблеми та інструменти їх вирішення. Розглянуто базову архітектуру запропонованої системи та модулі, що відповідають за генерацію анімації та її обробку на графічних процесорах.
Ключові слова: жестова мова, аватар, графічні рушії.

The article suggests an approach to the software system development, dedicated to the synthesis and the animation of the gestures and facial expressions of the emotions, based on the three-dimensional pattern of a human body. It describes main requirements for the system, analyzing basic problems and the implements for their solution. The article shows a basic architecture of the system, emphasizing in details its modules, especially these ones that are responsible of the generation of the animation, its structure, the algorithm of gestures' combination and their graphical processing.
Key words: sign language, avatar, graphics engines.

В статье предлагается подход к разработке программной системы для синтеза и анимации жестов, эмоционально-мимических проявлений на лице человека с использованием трехмерных моделей тела человека. Описываются основные требования к системе и инструменты их решения. Рассмотрены базовая архитектура предложенной системы и модули, отвечающие за генерацию анимации, а также ее обработку на графических процессорах.
Ключевые слова: жестовый язык, аватар, графические движки.

Вступ

Важливим етапом розвитку графічних процесорів та посилення їх значення для персональних комп'ютерів було створення високорівневих шейдерних мов, які дозволяли створювати складні алгоритми для відображення тривимірної графіки, підвищуючи реалістичність зображення [1]. Завдяки такому розвитку відеопроцесорів стала можливою розробка аватарів – віртуальних агентів для взаємодії людини з комп'ютером (human-computer interaction). Фундаментальними характеристиками аватарів є реалістичний вигляд та рухи, емоційна міміка та артикуляція, природнє промовляння слів та фраз. Основними проблемами при розробці аватарів є моделювання якісної тривимірної моделі, анімація реалістичних рухів, виразів обличчя, артикуляції, синтез природньої мови, синхронізація анімації зі звуком. Аватари використовуються в різних сферах, особливо в галузі розробки програмного забезпечення для людей з вадами слуху [2], [3]. У даній роботі розглядаються проблеми та особливості реалізації аватара для відтворення української жестової мови для побудови системи інтерактивного навчання жестовій мові.

У суспільстві важливою є проблема полегшення інтеграції людей з фізичними вадами. Це також стосується людей з вадами слуху, глухонімих. Такі люди переважно спілкуються жестовою мовою. Відмінності між жестовими та звуковими мовами визначає канал, яким передається інформація. У жестових мовах інформація кодується рухами рук, тіла, обличчя, очей і сприймається за допомогою зору. Це визначає фундаментальні властивості жестових мов:

1) провідну роль грає простір навколо людини, що говорить;
2) елементи жесту виконуються і сприймаються одночасно, в той час коли звукове мовлення поступово досягає вуха людини. Це дозволяє кодувати більший обсяг інформації порівняно зі словом усного мовлення.

Основною структурною одиницею жестової мови є жест. Жест у жестових мовах є аналогом звукового слова, тому деякі лінгвістичні методики можна успішно застосувати і до нього. Але жесту притаманні унікальні властивості, для опису яких такі методики не підходять, наприклад, фіксований порядок для визначення граматичного слова не потрібний, оскільки граматичні елементи в жестах виконуються одночасно [4].

У роботі [4] виділено три параметри, необхідні для опису структури жесту:

1) місце виконання жесту по відношенню до тіла людини, що промовляє;
2) форма кисті руки, що виконує жест;
3) траєкторія руху руки.

Відзначимо, що жестова мова є справжньою мовою з усіма її складнощами, також вона принципово і фундаментально відрізняється від розмовної мови правилами, принципами побудови речень і т.д. Для вирішення проблеми швидкого і ефективного вивчення жестової мови необхідне створення технологій для комп'ютеризації цього навчання в силу доступності та сучасності такого підходу.

Основні вимоги до системи синтезу жестів

Програмний комплекс для інтерактивного навчання українській жестовій мові складається з декількох підсистем, що відповідають за анімацію, звук, розпізнавання тощо. Найбільш важливою є підсистема – жестівник, що відповідає за реалістичну анімацію жестів, з одночасною анімацією артикуляції та різних емоційних проявів і звуковим супроводженням [5]. Основними архітектурними вимогами для такої системи є:

1) обчислення в режимі реального часу;
2) підтримка сучасних підходів і технологій тривимірної графіки для реалістично-

го зображення аватара;

3) кросплатформеність;
4) модульна структура;
5) організація бази жестів з великою кількістю даних.

Надзвичайно важливо, щоб система могла підтримувати обчислення в режимі реального часу, оскільки вона є досить ресурсозатратною. Для реалістичного відображення аватара, його рухів, емоційних проявів необхідно використовувати тривимірні моделі тіла людини з великою кількістю полігонів та скелетом, що складається з більш ніж 80 кісток. Також важливо, щоб тривимірний аватар виглядав максимально реалістично, був схожий на звичайну людину.

Ці завдання можливо вирішити за допомогою виконання деяких частин програми на графічному процесорі за допомогою програмованих шейдерів. Шейдер (англ. Shader) – це програма для одного зі ступенів графічного конвеєра, що використовується в тривимірній графіці для визначення остаточних параметрів об'єкта чи зображення. Вона може містити у собі довільної складності описи поглинання та розсіювання світла, накладення текстури, віддзеркалення і заломлення, затінення, зміщення поверхні і ефекти пост-обробки [1]. Програмовані шейдери гнучкі та ефективні. Складні на вигляд поверхні можуть бути візуалізовані за допомогою простих геометричних форм.

Характеристика основних графічних рушіїв

Важливим етапом у розробці системи є аналіз наявних технологій для відображення тривимірної графіки, так званих графічних рушіїв, основним завданням яких є візуалізація тривимірної графіки. Основними критеріями для таких рушіїв є відкритий доступ, підтримка програмних шейдерів на мовах GLSL, HLSL, CG, підтримка DirectX та OpenGL та можливість змінити графічну підсистему без зміни програмного коду, висока оптимізація, підтримка експорту тривимірних моделей із спеціалізованих програм. Також для реалістичного відображення тривимірної моделі тіла людини важлива підтримка різних типів освітлення, можливість по-різному відбивати світло від різних поверхонь, можливість одночасного відображення кількох анімацій.

Наведеним критеріям повністю відповідають три графічних рушії – Irrlight, Ogre, та Unity. Окрім того, вони є кросплатформеними, підтримують як OpenGL так і DirectX графічні бібліотеки, що в майбутньому може дозволити без проблем портувати систему на інші платформи. Основними перевагами Irrlight є його проста і прозора архітектура, невеликий, але достатній основний функціонал, сумісність з деякими аудіобібліотеками [6]. Ogre має більш складну архітектуру, потужний функціонал для управління матеріалами, підтримку позиційної анімації (pose animation), систему генерації шейдерів у реальному часі на основі скриптів матеріалів. Також великою перевагою Ogre є його гнучкість та можливість інтеграції з фізичними рушіями та аудіобібліотеками. Обидві системи мають досить розвинену спільноту, яка надає підтримку в разі проблем з їх використанням [7]. Ogre також підтримує мобільну систему iOS, але, на жаль, не підтримує мобільної системи Android без додаткових модифікацій. Unity ж, у свою чергу, є не графічним, а ігровим рушієм, тобто він має вбудовану підтримку звуку, фізики, мережі та інших речей, необхідних для тривимірних ігор. Водночас він є простим у користуванні та підтримує мобільні системи, такі як Android та iOS, що в час стрімкого зросту мобільних технологій є дуже важливим. Основним недоліком Unity є те, що це система з закритим кодом і безкоштовною є

лише урізана версія її для персональних комп'ютерів, для мобільних систем Unity є платною [8]. Описані характеристики наведені в табл. 1.

Таблиця 1 – Характеристики системи

	Ogre	Irrlight	Unity
Підтримка шейдерів	+	+	+
Підтримка DirectX/ Opengl	+	+	+
Кросплатформеність на ПК	+	+	+
Підтримка мобільних пристроїв	Тільки iOS	-	+
Підтримка звуку	Сторонні бібліотеки	Сторонні бібліотеки	Вбудована
Складність рушія	Середня	Легка	Складна
Бесплатність/Відкритість	+	+	-

Архітектура та деталі реалізації модулів програмної системи

Для вирішення проблем вивчення жестової мови програмна система повинна складатись з таких модулів, як:

1) звуковий генератор;
2) генератор міміки та емоційних проявів;
3) генератор анімації рухів та жестів;
4) компонента для синхронізації звуку та міміки;
5) графічний рушій;
6) звукова підсистема.

Кожен з цих модулів повинен бути незалежним та мати можливість бути заміненим іншою реалізацією без зміни решти компонент. Схема архітектури показана на рис. 1.

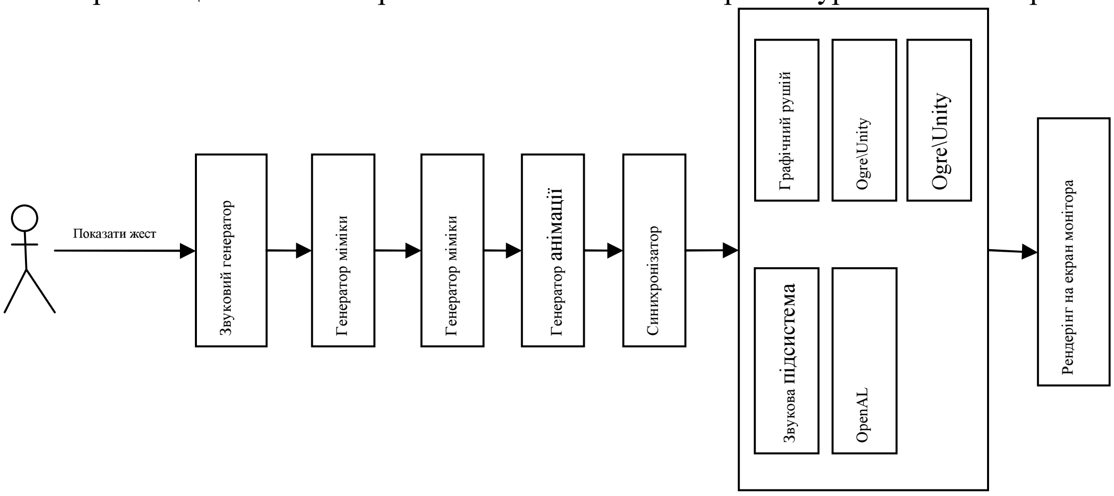


Рисунок 1 – Схема архітектури модуля

Використовуючи Ogre 3D, було реалізовано модулі 3 та 5 та побудовано програмне забезпечення (рис. 2) для відображення тривимірної моделі з простою скелетною анімацією, керуванням камерою та освітленням.

У системі було реалізовано 3 програмних шейдери:

1) шейдер для підтримки карт нормалей, що дозволить досягти більш високої деталізації аватара, не збільшуючи полігональність вхідного меша;

2) шейдер для підтримки карт відображень для більш реалістичного відтворення і відбиття освітлення;

3) шейдер для підтримки обрахунку скелетної анімації на графічному процесорі, що дозволив прискорити обрахунок цієї ресурсомісткої операції. FPS відтворення зріс від 30 до 120 за рахунок цього шейдера. Тестування проводилось на відеопроцесорі NVIDIA GeForce G102M.

Розглянемо детальніше структуру генератора анімації рухів. Він складається з двох основних компонентів – контролерів анімації: контролера анімації жесту та контролера анімації стану спокою.

Загальну схема контролерів анімації подано на рис. 2. Контролер анімації стану спокою поєднує роботу кількох контролерів одночасно (зміна напрямку зору; мікрорухи тулуба, зміна пози; мікрорухи головою; імітація дихання). Функціонально робота деяких контролерів є взаємозалежною (анімація руху очей і рухи голови) або взаємозамінною.

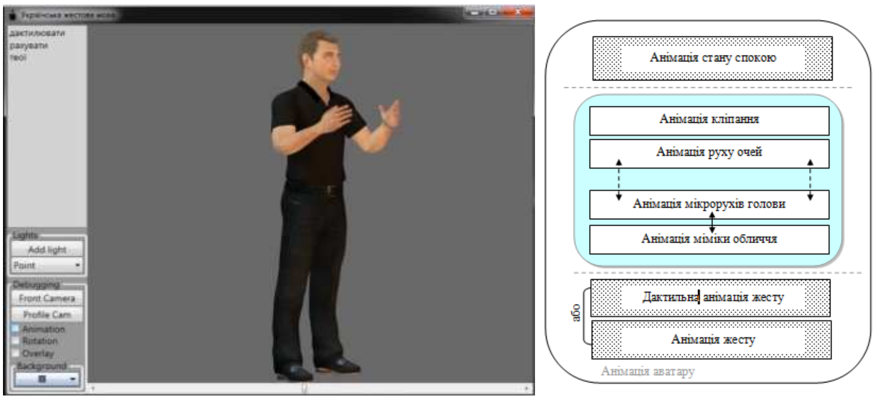


Рисунок 2 – Робоче вікно тестового програмного забезпечення

Важливим аспектом роботи системи є підсистема для управління анімаціями та переходами між ними. Було розроблено наступну модель, що підтримує три типи переходів між анімаціями:

1) моментальне завершення першої анімації та старт другої;

2) плавний перехід від першої анімації до другої;

3) очікування закінчення першої анімації і після цього старт другої.

Оскільки перший та третій пункти є досить простими, зупинимось детальніше на другому пункті.

Для забезпечення плавного переходу між двома анімаціями застосовувався наступний алгоритм. Спочатку була поставлена умова, що наступна анімація буде стартувати за 30% часу до закінчення стартової анімації. Після цього для кожної анімації встановлювався свій ваговий коефіцієнт, що визначався за наступною формулою: $weight_1 = timeLeft/(0.3*duration)$, $weight_2 = 1 - weight_1$, де $weight_1$ – ваговий коефіцієнт першої анімації, timeLeft – час, що залишився до кінця анімації, duration – довжина анімації.

Ще одним важливим аспектом системи є рухи обличчя: міміка, артикуляція, та емо-

ційні прояви. Для анімації обличчя переважно використовується морфна (morph target) або позиційна (pose) анімація. Скелетну анімацію важко використати для цієї задачі, оскільки існує багато різних комбінацій візем та мімічних проявів, що мають відображатись на обличчі одночасно. Морфна та позиційна анімація у цьому випадку підходять набагато краще, але морфна є більш затратною, оскільки для кожного фіксованого стану обличчя потрібно зберігати та обробляти позиції усіх точок обличчя, в той час як для позиційної анімації використовуються відхилення від стартового положення, і якщо воно нульове, тоді ця точка не обробляється. Водночас використовуючи позиційну анімацію, набагато легше змішувати різні типи анімацій, наприклад конкретну візему з певним емоційним станом. У побудованій системі для відтворення рухів обличчя планується використовувати саме позиційну анімацію.

Висновки

Отже, у даній роботі було досліджено особливості жестових мов та української жестової мови зокрема, виділено її основні особливості. Були проаналізовані основні вимоги до систем комп'ютеризації навчання жестових мов, розроблено базову високорівневу архітектуру для реалізації підсистеми синтезу жестової мови. Розглянуто основні інструменти для реалізації програмного забезпечення – сучасні бібліотеки для роботи з тривимірною графікою та звуком, реалізовані два основні модулі – генератор анімації та графічний рушій на основі Ogre, та певний користувацький інтерфейс для цих модулів.

Подальші дослідження будуть спрямовані на програмну реалізацію модуля для анімації міміки та емоційних проявів і її інтеграцію в наявну систему, створення програмного шейдера для проведення обрахунків цієї анімації на графічному процесорі. Водночас для реалістичного відображення очей необхідно додати до системи шейдер, що відповідатиме за відображення зовнішнього світу в очах аватара.

Література

1. Боресков А. Разработка и отладка шейдеров / Боресков А. – С.-Петербург : БХВ, 2006. – 496 с.
2. Smith R. HCI for the Deaf community: Developing human-like avatars for sign language synthesis / R. Smith, S. Morrissey, H. Somers // Proccedings of the 4th Irish Human Computer Interaction Conference. – 2 – September 2010, Dublin. – P. 129-136.
3. Hurdich J. Utilizing Lifelike, 3D Animated SigningAvatar Characters for the Instruction of K-12 Deaf Learners / J. Hurdich // Exploring Instructional and Access Technologies. Abstracts of International Symposium. – 23 – 25 June 2008. – New York : RIT, 2008. –P. 40-41.
4. Прозорова Е.В. Российский жестовый язык как предмет лингвистического исследования / Е.В. Прозорова // Вопросы языкознания. – М., 2007. – № 1. – С. 44-61.
5. Кривонос Ю.Г. Інформаційна технологія для моделювання української мови жестів / Ю.Г. Кривонос, Ю.В. Крак О.В. Бармак [та ін.] // Штучний інтелект. – 2009. – № 3. – С. 186-197.
6. [Електронний ресурс] Irrlicht. – Режим доступу : http://irrlicht.sourceforge.net/docu.
7. [Електрониий ресурс] Ogre. – Режим доступу : http://ogre3d.org/tikiwiki/.
8. [Електронний ресурс] Unity. – Режим доступу : http://unity3d.com/learn/.

Literatura

1. Boreskov A. Develoment and debugging of shaders / Boreskov A. – S.-Peterburg : BHV, 2006. – 496 p.
2. Smith R. HCI for the Deaf community: Developing human-like avatars for sign language synthesis / R. Smith, S. Morrissey, H. Somers // Proccedings of the 4th Irish Human Computer Interaction Conference. – 2 – 3 September 2010, Dublin. – P. 129-136.

3. Hurdich J. Utilizing Lifelike, 3D Animated SigningAvatar Characters for the Instruction of K-12 Deaf Learners / J. Hurdich // Exploring Instructional and Access Technologies. Abstracts of International Symposium. – 23 – 25 June 2008, New York : RIT. – P. 40-41.
4. Prozorova E.V. Russian sign languages as the subject of linguistic research / E.V. Prozorova // Problems of linguistic. – M., 2007. – № 1. – P. 44-61.
5. Kryvonos Yu.H. Information technology for synthesis Ukrainian sign language / Yu.H. Kryvonos, Yu.V.Krak, O.V. Barmak // Artificial intelligence. – 2009. – № 3. – P. 186-197.
6. Web resource Irrlicht. Access : http://irrlicht.sourceforge.net/docu.
7. Web resource Ogre. Access : http://ogre3d.org/tikiwiki/.
8. Web resource Unity. Access : http://unity3d.com/learn/.

RESUME

Ju.V. Krak, A.S. Ternov, M.P. Lisniak

The Development of the Architecture and Basic Implements of the Computer Animation for Sign Language System Synthesis Creation

Important problem of the integration of people with the disabilities is investigated. The problem applies both to the physical defects and hearing and speech impairments. Due to the accelerated development of the new technologies, the process of the communication becomes easier and opener. It allows the creation of the virtual agents (so-called «avatars») that facilitate the human interaction with the computer and promote the development of the software for the sign language teaching.

The research implies an approach to the development of the software system which includes the process of pronunciation, synthesis and animation of the gestures and emotionally-based facial expressions with the help of the three-dimensional model of the human body. It analyses basic characteristics of the gesture languages and emphasizes on the main requirements for the virtual «gesturer». The paper includes comparative characteristics of the modern researches, connected with the three-dimensional graphics. Moreover, it suggests the creation of the basic multi-architecture with the main purpose of the implementation of the subsystem of sign language. It describes in details the peculiarities of the animation generator and depicts the algorithm of the gestures' transition. The synthesis of the emotions and facial expressions is the important part of the system. The research suggests basic methods of the solution of this problem. Moreover, it contains the description of the software implementation of certain modules of the system.

Стаття надійшла до редакції 09.04.2013.